

Hardware Description Language Design of Σ - Δ Fractional-N Phase-Locked Loop for Wireless Applications

Ahmed El Oualkadi and Abdellah Ait Ouahman

Abstract—This paper discusses a systematic design of a Σ - Δ fractional-N Phase-Locked Loop based on HDL behavioral modeling. The proposed design consists in describing the mixed behavior of this PLL architecture starting from the specifications of each building block. The HDL models of critical PLL blocks have been described in VHDL-AMS to predict the different specifications of the PLL. The effect of different noise sources has been efficiently introduced to study the PLL system performances. The obtained results are compared with transistor-level simulations to validate the effectiveness of the proposed models for wireless applications in the frequency range around 2.45 GHz.

Keywords—Phase-locked loop, frequency synthesizer; fractional-N PLL; Σ - Δ modulator; HDL models.

I. INTRODUCTION

THE wireless communication market is undergoing a major expansion with the deployment of new technologies and standards opening the prospect of significant impacts in many application areas (security, health, automobile, environment, food security, manufacturing, telecommunications, robotics...). The emerging wireless technologies require architectures with reduced complexity, cost and power consumption; however, they require more accuracy and performance of specific circuits.

This evolution pushes designers both to find new architectures of circuits and systems which can offer high-performance, low-cost and low power consumption, and also to use new CAD (Computer-Assisted Design) methodologies able to model the mixed-mode behavior (analog / digital) of these systems. Currently, the hardware description language is widely applied in the design of mixed-signal circuits. Indeed, the VHDL-AMS standard allows the implementation of the top-down hierarchical approach for analog and mixed systems [1-3]. Therefore, it can be used straightforwardly for behavioral modeling and design of a phase locked loop (PLL), which is a key element for any wireless communication system.

Ahmed El Oualkadi is with the National School of Applied Sciences of Tangier, Route Ziaten, BP 1818, Tanger principale 90000, Morocco. (phone: +212669727216; fax: +212539393744; e-mail: ahmed.eloualkadi@ieec.org).

Abdellah Ait Ouahman is with the National School of Applied Sciences of Marrakech, Avenue Abdelkrim Elkhattabi, B.P. 575, Guéliz, Marrakech, Morocco.

Standard PLL frequency synthesizers with integer-N dividers have difficulties in meeting various specifications due to their fundamental tradeoffs between loop bandwidth and channel spacing. The fractional-N technique offers wide bandwidth with narrow channel spacing and alleviates phase-locked loop (PLL) design constraints for phase noise. The Σ - Δ fractional-N PLL [4-5] is indeed attractive for agile frequency synthesis or direct modulation. This architecture can still meet requirements such as low-power consumption and simple topology, and is suitable for high-level integration. The design of fractional-N PLL synthesizers, however, requires an iterative design process due to the large set of system parameters that must be optimized to achieve the desired phase noise, settling time, and fractional spur rejection. In addition, a Σ - Δ modulator used to instantaneously alter the feedback division modulus introduces excessive phase noise and fractional spurs. A behavioral level simulator is required to reduce the design turnaround time as well as assess the phase noise contribution and fractional spur rejection of the Σ - Δ modulator before the physical design phase. The need for a behavioral level simulator is strengthened by the characteristic that both the PLL and the Σ - Δ modulator are nonlinear systems. In the literature many papers have studied the implementation of the behavioral models of classical PLL systems and Σ - Δ synthesizers [6-8]. However, there are few works that show the full analysis and design of Σ - Δ synthesizers using behavioral modeling. In this paper, a Σ - Δ fractional-N PLL is modeled using VHDL-AMS and synthesized for a wireless application. For this study, many HDL models have been studied and tested for different PLL blocks. The VCO and Σ - Δ modulator are the major blocks that affect the PLL phase noise. These blocks are efficiently described and simulated in VHDL-AMS to estimate the PLL phase noise. The proposed PLL models use ELDO script [10] mixed with VHDL-AMS models which allow to simulate some PLL blocks at transistor-level and others at behavioral level. Several jitter noise sources are studied based on different noise models [11] and included into the PLL model to investigate non-ideal effects. These mixed behavioral models enable a fast simulation of the Σ - Δ synthesizer and an accurate phase noise prediction. A comparison with transistor-level simulations

validates the proposed models.

II. BEHAVIORAL MODELS OF PLL BUILDING BLOCKS

The architecture of the fractional-N PLL frequency synthesizer is shown in Fig. 1. It consists of a phase-frequency detector (PFD), a charge pump loop filter (CP & LF), a voltage controlled oscillator (VCO), an N/N+1 frequency divider, and an all-digital Σ - Δ modulator.

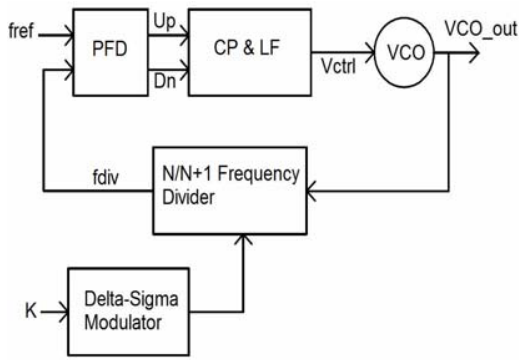


Fig. 1 Fractional-N PLL frequency synthesizer.

The static input word K is processed by a Σ - Δ modulator to produce an encoded oversampled sequence. This sequence is used to alter the division modulus of a multi-modulus divider in the feedback loop. Essentially, the average value of the encoded Σ - Δ output is equal to the DC input word K , resulting in an output frequency at a fractional multiple of the reference frequency. In the following subsections, we will introduce the modeling concerns of each block briefly.

A. Phase-Frequency Detector

Fig. 2 shows the conventional PFD structure and its ideal behaviors. This block detects the phase difference between the reference clock (f_{ref}) and the feedback clock (f_{div}).

In the top-down modeling, it is only described as a zero-delay three-state block that deals with the three situations shown in Fig. 2.

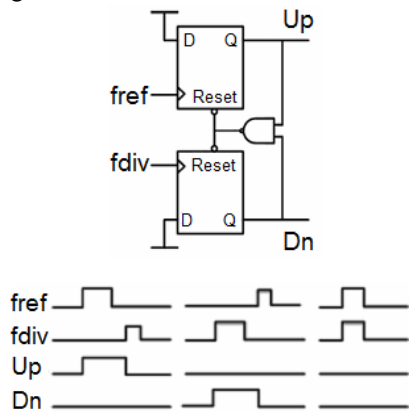


Fig. 2 Typical structure of PFD.

In the appendix, Listing 1. shows a part of initial VHDL-

AMS code of the frequency phase detector (FPD) model.

B. Charge Pump and Loop Filter

In the behavioral level, CP is equivalent to a pair of current sources and switches as shown in Fig. 3.

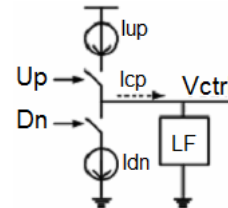


Fig. 3 Typical structure of CP and LF.

The typical loop filter is a low-pass filter composed of RC components. In traditional approaches, the behavioral model only describes the transfer function of the loop filter or keeps the transistor-level descriptions [12]. As mentioned in [13], using transfer function only is not enough for accurate behavioral simulation. However, directly using the values of the RC components appearing in the specification may not be so accurate because the equivalent RC values often have variance after layout due to parasitic effects.

C. Voltage Controlled Oscillator

The behavioral model of VCO typically describes the relationship between input control voltage and output frequency. The range of input operation voltage, the relative output frequency range and the VCO gain are the critical characteristics. In the top-down modeling approach, these parameters are obtained from the design specifications.

A complementary differential CMOS LC tuned VCO model has been used for transistor-level simulations (Fig. 4). For minimum power consumption and maximum output swing, both the cross-coupled NMOS-transistor and PMOS-transistor generate a negative resistance that compensates the loss of LC tank [14-15].

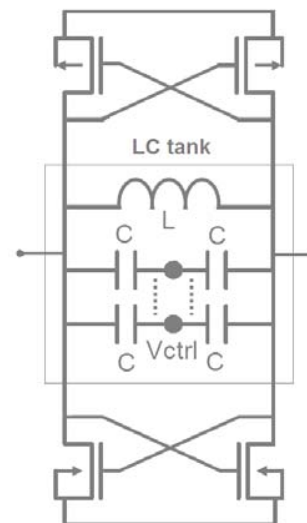


Fig. 4 Schematic diagram of the VCO.

D. $\Sigma\Delta$ Modulator and Frequency Divider

The $\Sigma\Delta$ modulator is a key block in the PLL used to produce the fractional part of the division ratio [16]. Fig. 5 shows the architecture of a 3rd order MASH $\Sigma\Delta$ modulator obtained by cascading three stages of 1st order $\Sigma\Delta$ modulator. The quantization error of every stage is injected to the next one. The corresponding quantized divider can be expressed as

$$N_3(z) = F(z) + R_3(z) (1-z^{-1})^3 \quad (1)$$

where $F(z)$ is the fractional input signal, and the last term represents the quantization noise, which only relates to the 3rd stage quantization noise $R_3(z)$ because the 1st and 2nd stage noises are eliminated in this structure [16].

The frequency divider is often treated as a pure digital block. The timing informations, such as delay time and output transition time, are the critical factors of this block.

The $\Sigma\Delta$ modulator can be clocked by either the reference clock signal or by the divider output, although using the divider output signal is reported to yield better performance. In this application, it is clocked by the divider output.

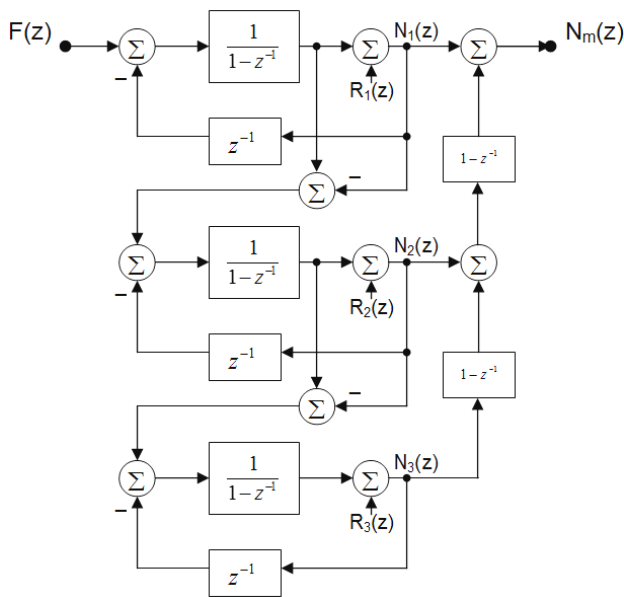


Fig. 5 3rd order MASH $\Sigma\Delta$ modulator structure

Listing 2, in the appendix, shows a part of VHDL-AMS model that combines together the VCO, the divider and the 3rd order MASH $\Sigma\Delta$ modulator into a single model.

III. PLL SYSTEM-LEVEL DESIGN

In order to validate the ability of VHDL-AMS to successfully describe the $\Sigma\Delta$ fractional-N PLL performance, a common and complex mixed-signal model has been developed. By using VHDL-AMS, the architecture of each block has been defined and simulated. Fig. 6 shows the

modeling approach used. The CP and PFD are modeled in VHDL-AMS. The loop filter is still modeled in Eldo (only R and C components). The VCO, divider and $\Sigma\Delta$ modulator are lumped into a single model, also in VHDL-AMS. Merging the VCO and the divider into a single model (Listing. 1) allows to avoid to explicitly generate the VCO output signal at a few GigaHertz. When using time-domain simulation, this modeling technique is the only way to obtain a reasonable CPU time.

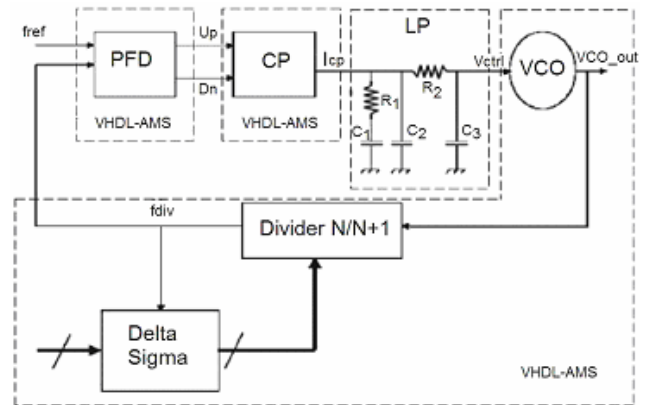


Fig. 6 $\Sigma\Delta$ fractional-N PLL modeling approach.

As it has been noted in the previous sections, the PLL models described at transistor-level are mixed with VHDL-AMS behavioral models which allow to simulate some PLL blocks at transistor-level and others at high description level. Fig. 7 shows the mixed-signal design flow proposed for this study.

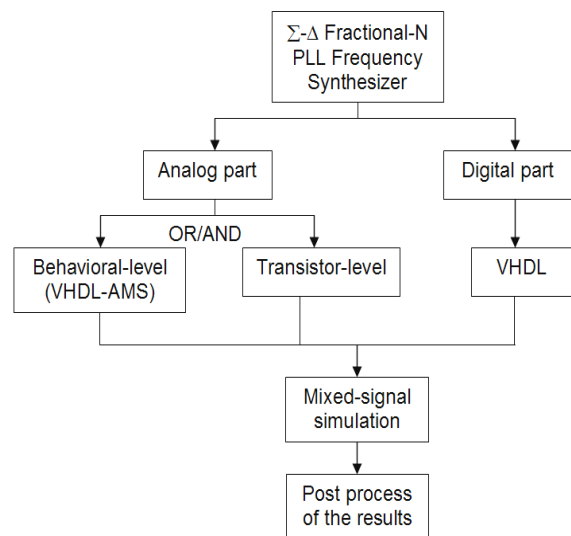


Fig. 7 Proposed mixed-signal design flow.

A. PLL Specifications

The loop filter is an important block to be optimized for reaching the target PLL bandwidth, phase margin and $\Sigma\Delta$

noise suppression. For simplicity, a 3rd order loop filter has been used in this study.

Fig. 8 shows the architecture of this loop filter. There are three capacitors and two resistors. C_1 produces the first pole at the origin for the type-II PLL. C_1 and R_1 are used to generate a zero for loop stability. C_2 is used to smooth the control voltage ripples and to generate the second pole. R_2 and C_3 are used to generate the third pole to further suppress reference spurs and the high-frequency phase noise in the PLL.

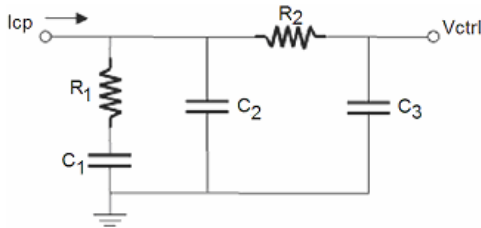


Fig. 8 3rd order passive loop filter for charge-pump PLL.

The use of a higher-order loop filter, however, requires careful design consideration, as the PLL is prone to instability. The average current-to-voltage transfer function of the loop filter is

$$F(s) = \frac{V_{out}(s)}{I_{avg}(s)} = \frac{D(s+1/\tau_1)}{\tau_2 \tau_3 s^3 + \left[\tau_2 + \left(\frac{D}{R_2} + 1 \right) \tau_3 \right] s^2 + \left(\frac{D\tau_3}{\tau_1 R_2} + 1 \right) s} \quad (2)$$

where $D=R_1C_1/(C_1+C_2)$, $\tau_1=R_1C_1$, $\tau_2=R_1C_1C_2/(C_1+C_2)$, and $\tau_3=R_2C_3$.

The open loop transfer function of the PLL can be determined from the following expression,

$$G(s) = \frac{K_d K_{vco} F(s)}{s N_{mean}} \quad (3)$$

where K_d and K_{vco} are the PFD constant and the VCO gain respectively. N_{mean} is the geometric mean of the maximum and minimum division ratio required to span the desired frequency band (in this case, $N_{mean} = (N + \text{fraction}) = 94.23$). Usually, f_{ref} and N_{mean} are defined from the target applications, while K_d and K_{vco} are optimized by the designers.

From (3), the PLL bandwidth and phase margin are decided by parameters such as reference frequency f_{ref} , divider ratio N_{mean} , PFD constant K_d , VCO gain K_{vco} and loop filter transfer function $F(s)$.

The open loop transfer function of the PLL has a zero located at $\omega_z=-1/\tau_1$, two poles at the origin, and two additional high frequency poles, denoted as ω_{p1} and ω_{p2} . Note that as long as $\omega_{p2} \gg \omega_{p1}$, the non-zero poles can be approximated by $\omega_{p1} \approx -1/\tau_2$ and $\omega_{p2} \approx -1/\tau_3$.

TABLE I
 $\Sigma\Delta$ FRACTIONAL-N PLL MAIN SPECIFICATIONS

Specifications	Values
PLL output frequency (f_{vco})	2.45 GHz
Reference frequency (f_{ref})	26 MHz
Channel spacing	200 KHz
VCO gain (K_{vco})	250 MHz/V
Nominal division ratio ($N_{mean}=N + \text{fraction}$)	94.23
Phase margin	60°

To achieve a 25 us settling time acceptable for a wireless application, the unity gain frequency of the open loop transfer function is located at $\omega_u=2\pi \cdot 200$ Krad/sec. 60° of phase margin is chosen to provide good settling behavior, dictating that $1/\tau_1=2\pi \cdot 50$ Krad/sec and $1/\tau_2=2\pi \cdot 800$ Krad/sec. The high frequency pole is located at $1/\tau_3=2\pi \cdot 6.6$ Mrad/sec to provide an additional 20 dB attenuation of the reference spurs. With these passive component values, the unity gain frequency is 199.18 kHz and the phase margin is 59.8°.

Based on many simulations using ADvance-MS from Mentor Graphics, the specifications of the $\Sigma\Delta$ fractional-N PLL have been established. Table I summarizes the PLL specifications for wireless application in the frequency range around 2.45 GHz. These closed-loop simulations take, for example, 2 minutes CPU time on a SunBlade 2500 machine.

B. Behavioral modeling of noise in PLL

It is very important to take into account the contribution of noise in the PLL building blocks, since this noise can directly affect the overall PLL performances which can distort the output spectrum of the PLL system. While it is difficult, for many reasons, to predict the phase noise in traditional circuit simulators [11], behavioral models can be used straightforwardly to predict the noise contribution in such systems. Indeed, Kundert [11] proposed an efficient approach to modeling phase noise in PLL compared to commercial simulators which take a long time to compute the system's dynamic response. Based on Kundert approach [11], many papers have described the behavioral modeling of noise in the PLL system [9, 11, 17, 18].

As mentioned in [11], there are two types of blocks in a PLL system, driven blocks and autonomous blocks. Each type exhibits a different type of jitter. Driven blocks, such as the PFD, CP, and divider give rise to phase modulation (PM jitter); autonomous blocks, such as the reference oscillator and VCO, to frequency modulation (FM jitter). This approach will be used in this study to simulate the PLL over-all noise.

IV. SYSTEM-LEVEL SIMULATIONS RESULTS

To simulate the performances of the $\Sigma\Delta$ fractional-N frequency synthesizer, all the HDL models must be connected together as described in Fig 6. Using the specifications given in Table I, it is possible to simulate the main characteristics of a $\Sigma\Delta$ fractional-N PLL for wireless application. These simulations have been performed by using PM (PFD) and FM

(VCO) noises sources. A jitter, equal to 2 ps, has indeed been introduced in HDL models to simulate the physical impact on PLL performances.

To start the closed-loop simulations, a 25 μ s transient simulation is performed to achieve the full locking process of the PLL.

Fig. 9 shows the transient analysis of the input control voltage (Vctrl) while the PLL is locking. The Σ - Δ fractional-N PLL has no steady-state solution, since the division ratio is changing all the time. Thus the control voltage, even when the PLL is locked, is changing continuously, modulating the VCO output frequency.

Fig. 10 shows the PLL output spectrum, with the carrier frequency 2.45 GHz, obtained by using a FFT algorithm, and Fig. 11 shows the output code of the Σ - Δ modulator. Since the Σ - Δ modulator is 3rd order, the dithered sequence is $\{-3, -2, -1, 0, 1, 2, 3\}$.

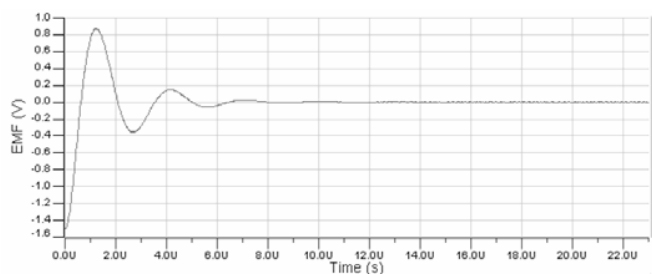


Fig. 9 Simulated dynamic of the closed-loop Σ - Δ fractional-N PLL.

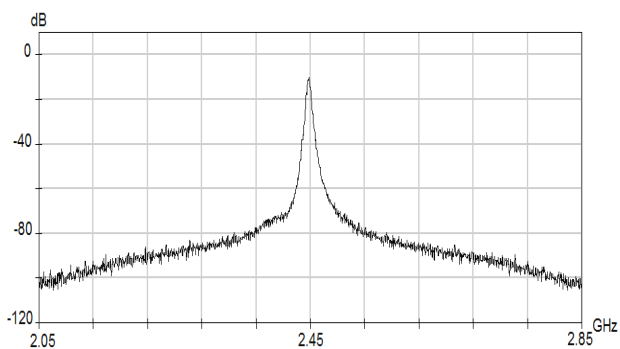


Fig. 10 Simulated PLL output spectrum.

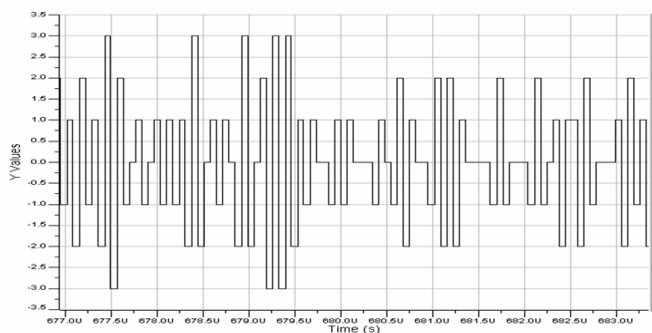


Fig. 11 Simulated output code of the 3rd order Σ - Δ modulator. The loop filter and the Σ - Δ modulator are important blocks

in the Σ - Δ fractional-N frequency synthesizer. To study the order impact of these blocks in the PLL design. Two architectures are simulated and compared using the 3rd order and the 4th order. Fig. 12 and Fig. 13 show the ideal (no noise sources used) simulations results obtained for these two cases.

Fig. 12 shows the output filter spectrum obtained with the proposed loop filter compared to that of a 4th order loop filter which uses the same architecture above with an additionally R-C branch at the output. This can filter more the high frequency spurs as shown in the figure.

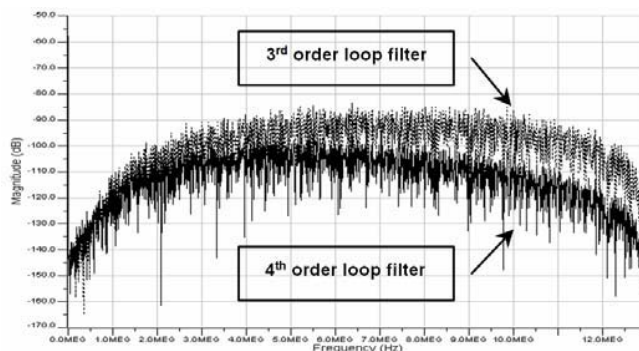


Fig. 12 Simulated output spectrums of the 3rd and 4th order filters.

Fig. 13 shows the output spectrums of the 3rd and 4th $\Delta\Sigma$ modulators. The 4th order modulator presents a low quantization noise in the bandwidth and more noise at high frequencies than the 3rd order modulator.

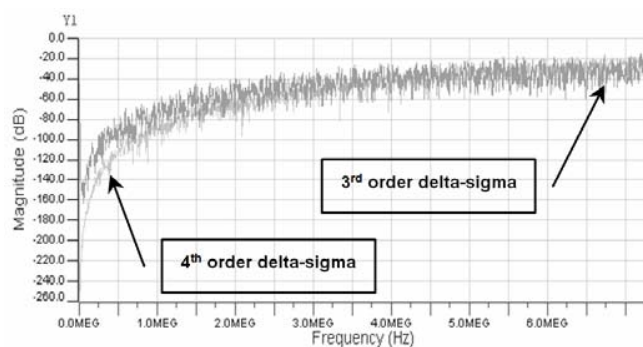


Fig. 13 Simulated output spectrums of the 3rd and 4th $\Delta\Sigma$ modulators.

From these simulations results (Fig. 12 and Fig. 13), it seems that a high order cannot give a huge improvement of the output response. Therefore, the choose of a 3rd order loop filter and a 3rd order Σ - Δ modulator is more important for wireless application since they can provide a low power and simple implementation of the fractional-N PLL building blocks.

Noise performance is the most critical specification for a frequency synthesizer. The PLL noise performance depends on all PLL blocks, but mainly on VCO phase noise.

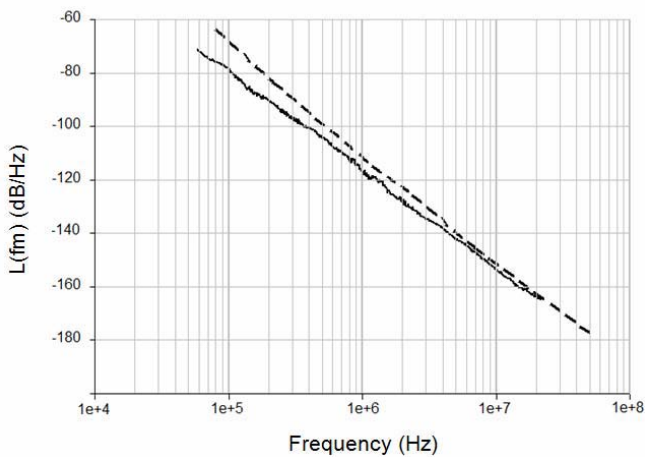


Fig. 14 Simulated VCO phase noise at different behavioral levels.

Fig. 14 shows the simulated VCO phase noise at different behavioral levels. The VCO transistor-level phase noise is obtained by a steady-state 'EldoRF' simulation, using 130 nm CMOS technology (Fig. 4). The behavioral simulation is obtained by using a FM jitter equal to 2 ps in the VCO HDL model. By comparing the two curves of Fig. 14, it seems that at low frequencies the transistor-level phase noise is dominant which can be explained by the $1/f$ noise presents in the CMOS technology and not taken into account in behavioral models.

Fig. 15 shows the simulated phase noise of the closed loop PLL. The simulation time is equal to 120 ms for one time step, while the total consumed CPU time is 4 hours. Fig. 15 shows the noise contribution of the different blocks of the PLL. The amount of this contribution depends on the level of jitter exhibited by the divider and PDF/CP.

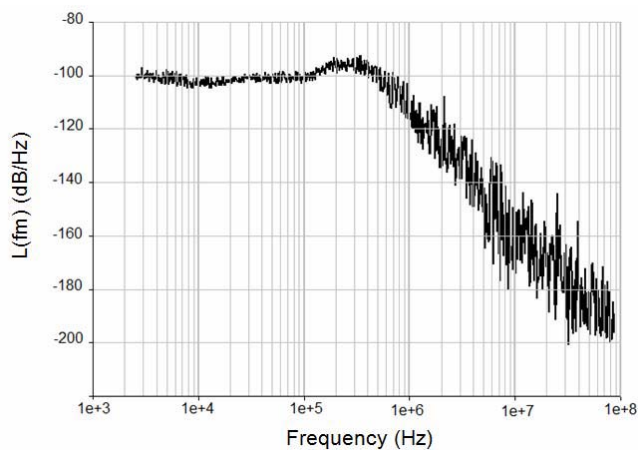


Fig. 15 Simulated phase noise of the Σ - Δ fractional-N PLL.

The phase noise is dominated by the VCO and Σ - Δ modulator in the range that goes from the cutoff frequency up to 10 MHz offset frequency, however the noise from PFD/CP and divider is dominating in the range of cutoff frequency. The reference oscillator noise contribution is clearly visible in

the lower frequency range. The fractional spurs out of the loop bandwidth is mainly caused by the Σ - Δ modulator.

V. CONCLUSION

The proposed paper has demonstrated the behavioral modeling and systematic mixed-design of $\Delta\Sigma$ fractional-N PLL using hardware description language VHDL-AMS. The behavioral modeling can provide a fast estimation of PLL performances compared to transistor-level simulation. These HDL behavioral models can be successfully mixed with some circuit blocks (transistor-level) to rapidly evaluate the contribution of each noise source and non-ideal element. This can help designers to test $\Delta\Sigma$ fractional-N PLLs, for a given wireless application, accurately within a minimum CPU time.

APPENDIX

Listing 1. A part of initial VHDL-AMS code of the frequency phase detector (FPD).

```
-- **** Entity ****
Entity FPD is
generic (pvdd : real := 5.0; -- positive supply voltage
pvss : real := 0.0; -- negative supply voltage
pthreshold : real := 2.5; -- threshold supply voltage
pdelay : real := 0.0; -- delay
prisetime : real := 1.0e-10; -- rise time
pfalltime : real := 1.0e-10; -- fall time
tresol : real := 1.0e-12; -- crossing resolution
port (terminal tref, tdiv, tup, tdn : electrical );
end entity FPD;
-- **** ARCHITECTURES ****
Architecture BEHAV of FPD is -- Branch quantities
quantity vup across iup through tup ;
quantity vdn across idn through tdn ;
quantity vref across tref ;
quantity vdiv across tdiv ;
-- Signals
signal sup, sdn : real := 0.0;
begin -- BEHAV
process (vref'above(pthreshold), vdiv'above(pthreshold))
variable vxup, vxdn : real := 0.0;
begin -- process
if (vref'above(pthreshold) ) then
if ( vref'above(pthreshold)'event) then
vxup := pvdd;
end if ;
end if;
if (vdiv'above(pthreshold) ) then
if (vdiv'above(pthreshold)'event) then
vxdn := pvdd;
end if ;
end if;
if ((vxup > pvdd/2.0) and (vxdn > pvdd/2.0)) then
vxup := pvss;
vxdn := pvss;
end if ;
sup <= vxup after pdelay*sec ;
sdn <= vxdn after pdelay*sec ;
end process;
break on sup, sdn ;
vup == sup'ramp(prisetime, pfalltime);
```

```

vdn == sdn'ramp(prisetime, pfalltime);
end architecture BEHAV;

```

Listing 2. A part of initial VHDL-AMS code that joins together the VCO, the 3rd order MASH Σ - Δ modulator and the divider into a single model.

```

-----
-- **** Entity ****
entity vco_fdiv_MASH is
  generic (fvco_vin0: real := 1.0e3; -- free running VCO output
  frequ
    Kvco : real := 1.0e3; -- vco gain
    vcod : real := 0.5; -- distortion on voltage-frequency
  characteristic
    int_div : real := 1.0; -- nominal division factor
    pvdd : real := 3.0; -- positive supply voltage
    pvss : real := 0.0; -- negative supply voltage
    num_bits : integer := 10; -- number of bits of the DS
  accumulators
    phalftref : time := 0fs;
    fraction : integer := 0 -- DS input code
  );
  port (terminal tin, tout, tdsclock : electrical;
    signal spy_ds, spy_f : out real := 0.0
  );
end entity vco_fdiv_MASH;
-- **** ARCHITECTURES ****
architecture behav_ds111 of vco_fdiv_MASH is
  -- branch quantities
  quantity vin across tin;
  quantity vout across iout through tout;
  quantity vdsclk across tdsclock;
  -- free quantities
  quantity xd, nper, arg, phase_total : real := 0.0;
  -- signals
  signal sdivfactor : real := int_div;
  signal Sdsout : integer := 0; -- DS output code
  -- constants
  constant ref_num_bits : integer := 2**num_bits;
  constant ref_time : real := 1.0/(fvco_vin0/int_div);
  constant vmid : real := (pvdd+pvss)/2.0;
  constant vamp : real := (pvdd-pvss)/2.0;
  constant two_pi_fvco_vin0 : real := math_2_pi * fvco_vin0;
  constant two_pi_Kvco : real := math_2_pi * Kvco;
begin
  -- SD output changes instantaneously division factor
  break on Sdsout;
  -- ** VCO & DIVIDER **
  -- distortion on vco transfer characteristic
  if (vcod > 0.0) use
    xd == vcod * tanh(vin/vcod); -- distorts input voltage
  else
    xd == vin;
  end use;
  -- compute divider output phase
  arg == (two_pi_fvco_vin0 + two_pi_Kvco *
  xd)/(int_div+real(sdsout));
  nper == trunc(arg/integ/math_2_pi);
  phase_total == arg/integ - real(nper)*math_2_pi;
  -- divider output signal
  vout == vmid + vamp * sin(phase_total);
  -- ** DS MASH 111 register-level **
DS_MASH111: process

```

```

-- registers
variable ne1, ne2, ne3, ne1_1, ne2_1, ne3_1, d3, d2, d1, d2_1, d3_1,
d3_2: integer := 0;
begin
  wait until vdsclk'above(vmid);
  -- register like coding for z^-1 blocks
  ne1_1 := ne1;
  ne2_1 := ne2;
  ne3_1 := ne3;
  d3_2 := d3_1;
  d3_1 := d3;
  d2_1 := d2;
  -- 1-bit modulators
  -- 1st stage
d1 := integer(sign(real(fraction + ne1_1 - ((fraction + ne1_1) mod
2**num_bits)))); --MSB
ne1 := (fraction + ne1_1) mod 2**num_bits; -- LSB
  -- 2nd stage
d2 := integer(sign(real(ne1 + ne2_1 - ((ne1 + ne2_1) mod
2**num_bits))));
ne2 := (ne1 + ne2_1) mod 2**num_bits;
  -- 3rd stage
d3 := integer(sign(real(ne2 + ne3_1 - ((ne2 + ne3_1) mod
2**num_bits))));
ne3 := (ne2 + ne3_1) mod 2**num_bits;
  -- noise-cancellation logic
Sdsout <= d1 + d2 - d2_1 + d3 - d3_1 - d3_1 + d3_2;
  spy_ds <= real(Sdsout);
  spy_f <= (fvco_vin0+Kvco*xd)/fvco_vin0;
end process DS_MASH111;
end architecture behav_ds111;

```

REFERENCES

- [1] Design Automation Standards Committee of the IEEE Computer Society, IEEE Standard VHDL Analog and Mixed Signal Extensions. - 314 pages, Doc., IEEE Std 1076.1-1999, 18 March 1999.
- [2] Gregory Peterson, Peter J. Ashenden, Darrell A. Teegarden, The System Designer's Guide to VHDL-AMS: Analog, Mixed-Signal, and Mixed-Technology Modeling, Morgan Kaufmann Publishers; 2002.
- [3] Y. Hervé, VHDL-AMS: applications et enjeux industriels, Dunod, Paris 2002.
- [4] T.A. Riley, M. A. Copeland. Delta-Sigma Modulation in Fractional-N Frequency Synthesis. In IEEE J. Solid State Circuits, vol. 28, pp. 553-559, May 1993.
- [5] M.H. Perrott, T.L. Tewksbury, and C.G. Sodini, A 27-mW CMOS fractional-N synthesizer using digital compensation for 2.5-Mb/s GFSK modulation. IEEE J. Solid-State Circuits, vol.32, no.12, pp.2048-2060, Dec. 1997.
- [6] M. Hinz, I. Konenkamp and E.H. Horneber. Behavioral Modeling and Simulation of Phase-Locked Loops for RF Front Ends. In IEEE Midwest Symp. On Circuits and Systems, pp. 194-197, Aug. 2000.
- [7] N. Milet-Lewis, G. Monnerie, A. Fakhfakh, and all. A VHDL-AMS library of RF blocks models. IEEE International Workshop on Behavioral Modeling and Simulation, 12 - 14, 2001.
- [8] M.H. Perrott, Fast and accurate behavioral simulation of fractional-N frequency synthesizers and other PLL/DLL circuits, Proc. 39th Design Automation Conf., pp.498-503, June 2002.
- [9] M. H. Perrott, M.D. Trott, and C. G. Sodini. A Modeling Approach for Σ - Δ Fractional-N Frequency Synthesizers Allowing Straightforward Noise Analysis. In IEEE Journal of Solid-State Circuits, Vol. 37, No. 8, pp. 1028-1038, Aug. 2002.
- [10] Eldo User's Manual. Mentor Graphics, 1998.
- [11] K. S. Kundert. Modeling and Simulation of Jitter in Phase-Locked Loops. Cadence Design Systems. San Jose, California, USA.
- [12] A. Mounir, A. Mostafa, and M. Fikry, Automatic behavioural model calibration for efficient PLL system verification. Design, Automation and Test in Europe Conference and Exhibition, pp.280-285, 2003.

- [13] B.A.A. Antao, F.M. El-Turky, and R.H. Leonowich, Behavioral modeling phase-locked loops for mixed-mode simulation. *Analog Integr. Circuits Signal Process.*, vol.10, pp.45–65, 1996.
- [14] T. H. Lee. *The Design of CMOS Radio-Frequency Integrated Circuits*. Cambridge University Press, pp 455-463, 1998.
- [15] M. Tiebout. Low-power low-phase-noise differentially tuned quadrature VCO design in standard CMOS. In *IEEE J. Solid State Circuits*, vol. 36, pp. 1018-1024, July 2001.
- [16] N. M. Filiol, T.A.D. Riley, C. Plett, and M.A. Copeland. An agile ISM band frequency synthesizer with built-in GMSK data modulation. In *IEEE Journal of Solid-State Circuits*, Vol. 33, No. 7, pp. 998-1008, July 1998.
- [17] K. Kundert. Predicting the Phase Noise and Jitter of PLL-Based Frequency Synthesizers. www.designers-guide.com, May 2003.
- [18] L. Yang, C. Wakayama and C. Richard Shi. Noise Aware Behavioral Modeling of the S-D Fractional-N Frequency Synthesizer. *Proc. Great Lakes Symp. on VLSI*, pp. 138-142, 2005.



Ahmed El Oualkadi received B.S. and M.S. degrees in physics and electronics from Abdelmalek Essaadi University, Tetuan, Morocco, in 1998 and 2000, respectively. He received Ph.D. degree in electronics from the University of Poitiers, France, in 2004. From 2001 to 2003, he was a research assistant in the Laboratoire d'Automatique et d'Informatique Industrielle – Ecole Supérieure d'Ingénieurs de Poitiers, Electronics & Electrostatics Research Unit, University of Poitiers, France. In 2004, he was an assistant professor at University Institute of Technology, Angoulême, France. During this period, he worked, in collaboration with EADS-TELECOM, on various European projects (CORMORAN & MULTIMODULES) which concern the nonlinear analysis & RF circuit design of switched-capacitor filters for radio-communication systems. In 2005, he was a research associate in the Université Catholique de Louvain, Microelectronics Laboratory, Louvain-la-Neuve, Belgium, where he worked on the design of low power high temperature circuits and systems, in SOI technology, for wireless communication especially for ZigBee (PHY IEEE 802.14.5) applications. During this period, he managed several European and regional projects (EUREKA, A 109 Witness, MEDEA+, CROTALE...) in the areas of wireless communication and sensor networking. Currently, he is an assistant professor in the Ecole Nationale des Sciences Appliquées of Tangier, Morocco. His main research interest is the analog, mixed-signal and RFIC design for wireless communication and embedded system applications. Since 2002, he is an active IEEE member associated to the Circuits & Systems Society where he is a reviewer of IEEE journal of circuits and systems and ISCAS conferences. He is author/co-author of more than 30 publications and communications in recognized journals and international IEEE conferences.



Abdellah Ait Ouahman received the doctorate thesis in Signal Processing from the University of Grenoble, France, in November 1981. His research was in Signal Processing and Telecommunication. Then he received Ph.D degree in Physics Sciences from the University of Sciences in Marrakech, Morocco, in 1992. He is now Professor and responsible of the Telecommunication and Informatic Networks Laboratory in the Faculty of Sciences Semlalia in Marrakech, Morocco and also is Director at ENSA (National School of Applied Sciences), Marrakech. His main research interests include the signal and image processing and coding, telecommunications and networking.